Inclusive Computing in Special Needs Classrooms: Designing for All

Zuzanna Lechelt[1], Yvonne Rogers[1], Nicola Yuill[2], Lena Nagl[2], Grazia Ragone[2], Nicolai Marquardt[1]
[1]University College London, UCLIC, Gower Street, London, UK
[2]University of Sussex, Children & Technology Lab, School of Psychology, UK
{susan.lechelt.15, y.rogers, n.marquardt}@ucl.ac.uk, nicolay@sussex.ac.uk, lena.nagl2@gmail.com, grazia.ragone@gmail.com

ABSTRACT
With a growing call for an increased emphasis on computing in school curricula, there is a need to make computing accessible to a diversity of learners. One potential approach is to extend the use of physical toolkits, which have been found to encourage collaboration, sustained engagement and effective learning in classrooms in general. However, little is known as to whether and how these benefits can be leveraged in special needs schools, where learners have a spectrum of distinct cognitive and social needs. Here, we investigate how introducing a physical toolkit can support learning about computing concepts for special education needs (SEN) students in their classroom. By tracing how the students' interactions—both with the physical toolkit and with each other—unfolded over time, we demonstrate how the design of both the form factor and the learning tasks embedded in a physical toolkit contribute to *collaboration*, *comprehension* and *engagement* when learning in mixed SEN classrooms.

Author Keywords
Digital fluency; computational thinking; physical interfaces; special needs education; computer-supported learning

ACM Classification Keywords
H.5.2. Information interfaces and presentation (e.g., HCI): User Interfaces, Evaluation

INTRODUCTION
The argument for getting *all* school-aged children to learn computing is now universally accepted. The benefits are assumed to be many; specifically, it is well documented that not only does learning computing teach students how to code and create digital content, but it also teaches a set of domain-general problem solving competencies that can be practically applied in many aspects of day-to-day life. These include the ability to break down problems into smaller parts, and to draw on both logic and creativity to figure out the best ways to solve them [19,39].

CHI 2018, April 21–26, 2018, Montreal, QC, Canada

ACM 978-1-4503-5620-6/18/04…$15.00
https://doi.org/10.1145/3173574.3174091

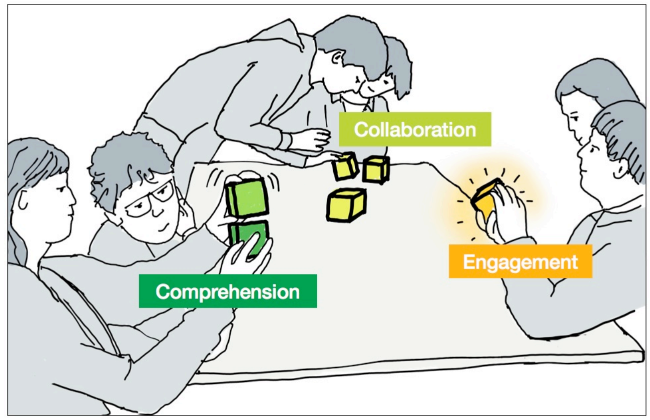

Figure 1: Special education needs students interacting with the Magic Cubes physical toolkit.

However, in debates about the best practices of teaching computing, little has been said about how to include learner groups that are often overlooked (for emerging work, see e.g., [16,33,35]). In particular, there has been little research on the best ways for teaching computing for mixed special education needs (SEN) settings. Researchers face a number of challenges and opportunities in these settings. For example, in special needs schools, the goal is foremost to teach holistic skills that will lead students to succeed in independent life and work. It is important that academic learning also supports the development of such related skills. In addition, in special needs schools, classrooms are often mixed; students are rarely grouped in classrooms according to their primary diagnosis, such as Autism Spectrum Disorder (ASD), general learning difficulties or sensory impairments. Rather, in mixed classroom settings, students with different profiles have both distinct needs and distinct strengths, often with a larger spread in abilities than in mainstream classrooms. This poses a challenge for researchers and teachers: how can the needs and strengths of students in a mixed SEN classroom be best supported to learn computing?

One promising approach is to use physical and tangible toolkits to help students learn about computing and coding.

However, so far the focus has been on mainstream settings. In these settings, tangible programming has been shown to give rise to more collaboration between learners than purely digital programming [15], and the realism and physical interaction afforded by physical interfaces has been suggested to lead to more engaging and embodied experiences [41]. What if these same properties could be tapped into and even other benefits discovered for students learning with SENs?

We are interested in whether the properties of physical toolkits can also lend themselves to helping SEN students collaborate more and harness their ability to think abstractly when learning about computing. Specifically, our research is concerned with addressing the following question: *how can the potential benefits of using physical interfaces for teaching computing concepts provide an experience that is collaborative, engaging and supports comprehension, for a spectrum of learners in a typical SEN school setting?* In answering this, it is important to consider the design of the learning task as well as the interface itself, and the roles of the teachers and key workers who work with SEN students in these settings.

By drawing on previous research on tangible and physical interfaces, as well as on the literature about mixed special needs groups, we designed and conducted a series of learning sessions during a school term using a physical toolkit—the Magic Cubes [8]—in a SEN classroom for students aged 16-19. The design of the sessions emphasized providing appropriate conceptual scaffolding as well as a range of learning tasks through a variety of discovery-based and coding activities. By qualitatively analyzing the students' learning pathways with the physical interface, as well as their subjective experiences during the sessions, we report on how the design of the learning activities and the form factor of the physical toolkit contribute to successful *collaboration*, *comprehension*, and *engagement* for a diversity of learners when learning about computing. We discuss the lessons learned and, in particular, the benefits accrued from both the design of the *technology* and the *learning task* for interventions that are able to accommodate a mixed SEN environment.

BACKGROUND

Special Education Needs (SEN)

14.4% of school-aged students in the UK are said to have special education needs [9]. In England alone, there are over one thousand government-funded and private SEN schools [9]. In these schools, learners often have a variety of special education needs, including Autism Spectrum Disorder (ASD), severe and moderate learning difficulties, as well as specific neurological impairments, such as acquired brain injury or sensory impairments.

Recently, in education research, there has been an increasing move away from assigning SEN students to rigid categories based on their primary diagnoses. Rather, terms including "learning difficulties" [11] and "developmental diversity" [6] have begun to be consciously adopted as part of a movement towards more inclusive education. These terms reflect a shift to a more social constructionist perspective on SEN [23], where learners' performance and potential is considered to be dynamic rather than fixed, and contingent on the level and type of support they receive in the learning environment. Through this perspective, the onus on academic achievement is shifted away from the students' difficulties and disabilities, and towards the support provided by the school, teacher and tools.

Although each specific special education need has its own profile, it has been suggested that as a group, learners with SEN face a number of similar key challenges. These include difficulty in dedicating sustained attention to the task at hand, and difficulty with understanding and recalling abstract concepts [11]. Additionally, especially learners with ASD face challenges with a number of processes related to collaboration, such as recognizing the other as a partner in interaction and building and sustaining joint awareness [13]. These challenges underpin all four factors addressed by SEN school curricula, which aim foremost to support learners' cognitive development; development of communication; physical, motor and sensory development; and emotional and social development [9].

Active, Constructive and Embodied Learning

When determining how best to design technology to support SEN learning, it is helpful to look to the learning sciences to operationalize how successful learning occurs. At the core of modern learning theory lies Piaget's constructivism, which posits that learning entails the incremental refinement of mental models, through which previously learned assumptions are continuously reorganized [22]. Crucially, this reorganization cannot occur passively, but must instead be active and reflective [1]. A number of theorists have further proposed how active and reflective learning can be supported through activity. Specifically, Vygotsky's social constructivist perspective [38] emphasizes the importance of dialogue when learning, as a way of verbally reflecting on and clarifying assumptions. Papert's constructionism [21], in turn, advocates the value of augmenting the learning process with *objects-to-think-with,* or concrete representations (physical or digital) of abstract concepts in the real world. Within HCI, Dourish's emphasis on cognition being embodied, rather than only situated in the brain, suggests the importance of using the body to create meaning when learning [2,10].

By providing a concrete and embodied way of exploring abstract concepts [40] as well as giving rise to collaborative activity [34], tangible and physical interfaces have much scope to support successful learning [20]. Moreover, the benefits of tangible and physical interfaces have been suggested to support the key learning challenges in SEN, specifically by providing multiple representations of abstract concepts, opportunities for physical manipulation, and through enabling collaboration [11]. However, although they have been explored in research for specific learning disabilities, and especially for students with ASD (e.g., [12]), work on introducing them to mixed SEN classrooms is still limited.

Moreover, when designing technologies for SEN classroom settings, it is important to consider the way in which they are presented, as well as the learning tasks with which they are used. An exploratory study of a SEN classroom [11] and a systematic literature review [6] have suggested that in mixed SEN settings, the introduction of novel technologies should: foster a sense of achievement through short and attainable learning tasks; scaffold learning tasks to enable students with differing abilities to succeed; provide instructions through multiple mediums (e.g., verbal and written) to support different types of learning; enable easy support from instructors; and provide opportunities for students to easily observe and collaborate with each other [6,11]. The goal of our study was to design an intervention that would utilize these strategies, and to test their efficacy when introducing a physical interface to teach computing in a mixed SEN classroom.

Learning Computing with Physical Interfaces

Learning about computing concepts is increasingly valued in school curricula for its ability to teach transferable computational thinking skills, and give rise to constructive, hands-on learning experiences [27,39]. Much research has been carried out to identify how abstract computing concepts [7] can best be brought down to a level that makes learning about them easy and fun for children and novices [26]. In particular, programming languages for children and novices, such as Scratch [28], have been extensively researched to explicate what features of both the programming language itself and the broader learning environment can enable successful learning.

In recent years, a variety of physical and tangible toolkits for learning computing have also been created. These come in many shapes and forms, from blocks that users can connect to create simple computational programs [5,14,35], to reprogrammable microcontrollers and systems-on-a-chip [3,4,42], to interfaces enabling discovery-based exploration of abstract hardware and systems concepts [17,29,40]. These toolkits often teach a range of computing concepts that extend beyond programming, including the functionalities of electronic hardware, and the connections between hardware and software. Moreover, it has been suggested that they can engender a more collaborative learning experience than desktop-based software [15] and afford more embodied and engaging learning experiences [41]. Analyses of students' perceptions of physical interfaces for learning computing have suggested that seeing abstract computing concepts translated to the real world makes them easier to understand [32]. There appears to be much potential for students with special needs to also benefit from these properties. However, the few studies that have been carried out have been largely for one type of special need, or for sessions in the lab (e.g., [37]). Here, we are interested in how the novel, physical formats can be explored by students with mixed abilities in a more naturalistic setting - their classroom – with which they are familiar and used to learning in.

STUDY DESIGN AND METHODOLOGY

The aim of our study was to assess the benefits of using physical toolkits for learning about computing concepts with diverse SEN students in a real world setting. Specifically, the goal was to investigate what factors of both the interface design and the learning task would support *collaboration, engagement with the content* and *comprehension of abstract computing concepts,* and in a mixed setting. To this end, we designed activities that tried to match the needs of the whole classroom, bearing in mind the needs of the individual students, as well as the central role of the key workers and teacher in a SEN classroom.

Participants

The study took place in a computing class at a special needs school in the UK. Eleven students aged 16-19, comprising 9 males and 2 females, participated in the study. This was a typical size of a classroom setting for special needs students. The preponderance of male students in the class may have been due to the fact that the school had a high ratio of male to female students, as well as them electively enrolling in the computing class and having a prior interest in computing. The students had a range of special needs (see Table 1). The most prevalent primary diagnosis was ASD (n=6), followed by moderate and specific learning difficulties (n=3), which is representative of UK SEN demographics [9]. The class had one main teacher, as well as two key workers (also a typical set-up), who supported the students with communication (e.g., through sign language) and learning tasks. Both the teacher and the key workers were present and actively involved in all sessions. The students chose their groups for the sessions. To run the studies and help with the activities, 3 to 4 other researchers were present in each session, each walking around the classroom and helping the groups when needed.

Name *	Gender	Group	Primary Diagnosis
Jason	M	G1	Autism Spectrum Disorder
Keith	M	G1	Acquired Brain Injury
David	M	G2	Autism Spectrum Disorder
Eric	M	G2	Specific Learning Difficulties/ Speech and Language
Ali	F	G2/ G3	Hearing Impairment/ Moderate Learning Difficulties
Curtis	M	G3	Autism Spectrum Disorder
Fabian	M	G3	Social, Emotional, Mental Health
Neil	M	G4	Autism Spectrum Disorder
Teddy	M	G4	Autism Spectrum Disorder
Lily	F	G5	Moderate Learning Difficulties
Gary	M	G5	Other, not specified

Table 1. Description of the students' profiles. *All names have been changed to protect the participants' anonymity.

Materials

The technology used during the intervention was the Magic Cubes toolkit [8]. This comprises physical, interactive

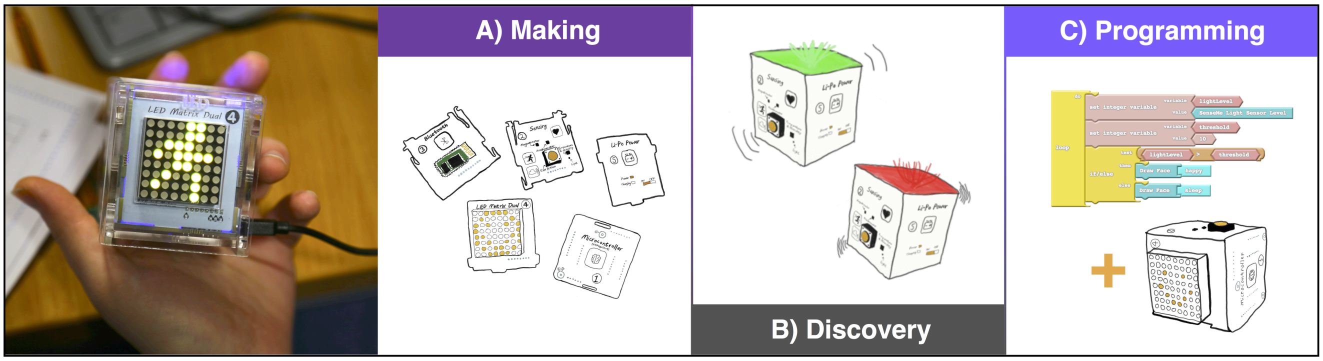

Figure 2. The types of learning tasks enabled by the Magic Cubes toolkit. (A) The toolkit as a flat printed circuit board before assembly. (B) Pre-programmed cubes can be explored through discovery-based tasks (e.g., the speed of shaking the cube changes the color of the neopixel light). (C) The cubes can be programmed using a visual, block-based programming language.

sensing cubes (see Figure 2) that can sense an assortment of data, including light level, temperature and acceleration. The data can then be visualized through an embedded LED multi-color light or 8*8 LED matrix. The Magic Cubes were chosen as the toolkit to use in our study as they have been shown to support a range of types of learning activities [8]. This enables us to analyze the effects of combining different learning tasks using the same interface form factor. Specifically, they can be assembled from a flat printed circuit board sheet [17] (Figure 2A). They can then be pre-programmed with sensor-actuator effects to be explored through guided discovery (Figure 2B). For example, shaking the cube can produce different colors depending on the speed of movement, or blowing hot air into the temperature sensor can produce a larger animation on the LED matrix. The cubes can also be creatively programmed by the students using a PC with an Arduino-based visual programming language (Figure 2C).

Procedure

The intervention was carried out through six weekly 90-minute sessions during the students' regular computing class timeslot. Prior to and throughout the intervention, we communicated with the class teacher about the demographics of the class and the specific needs and interests of the students, and integrated his responses into the planned learning tasks. We also communicated the planned learning tasks with the teacher before each session, in order to improve them, based on his feedback. The intervention as a whole aimed to cover the following computing concepts, chosen to be in line with the UK national computing curriculum [36] and the aims of the computing class that the students were enrolled in:

1. Understanding the functionality of core hardware components in a computer
2. Understanding the functionality of sensors and actuators
3. Understanding the functionality of wireless Bluetooth connectivity
4. Understanding and writing basic algorithms
5. Understanding and programming *if/else statements*
6. Understanding and programming *for loops*
7. Understanding and programming *bitmaps*

Before the researchers arrived at the first session, the teacher explained to the students what was going to happen and what they would be learning in the following six weeks. Ethical approval was obtained for the project; the parents of the students were informed of the project and gave their consent for their children to participate and for data to be recorded. At the beginning of the first session, the researchers were introduced. The students were informed about the purpose of the research, and it was explained that the videos would not be shared with anyone other than the researchers. The students were asked if they would like to take part in the research and whether they would mind being filmed, and all consented.

Throughout the intervention, the students were asked to work in pairs or groups of three. The students chose their own partners. This was done to encourage collaboration and dialogue while learning. Throughout the intervention, the majority of the students remained in the same pairs. There were two exceptions. In week 1, Fabian, a new student from Italy who had limited English fluency, worked with an Italian researcher, who helped him translate the verbal instructions. Later in the same session, he worked with Curtis and Ali (G3). From week 2 onward, Fabian worked only with Curtis. Ali, who was in a pair with Curtis (G3) in week 1, was absent for three sessions due to a conflicting personal appointment. From week 5, she joined a group with David and Eric (G2).

Session Design

During the study, three of the six sessions utilized the Magic Cubes toolkit. Each of these sessions was followed in the subsequent week with a toolkit-free task, designed to consolidate the concepts that were learned while using the toolkit. This was done to provide the students with opportunities to reflect on the computing concepts they had learned. In addition, it allowed us to shape the learning activities based on the observed needs and comprehension of the students.

The six sessions (Table 2) were planned by taking into account empirically-grounded design considerations from previous research on designing learning interventions for

SEN students [6,11] and more generally, on designing effective physical interfaces for learning [14,21,29,40]. These were: (i) capitalizing on embodied interaction to promote concrete, kinesthetic learning and collaboration between peers; (ii) enabling success for students of diverse abilities through short, attainable and conceptually scaffolded tasks; (iii) providing the students with instructions through multiple representations (verbal, visual and written); and (iv) providing opportunities for reflection on and consolidation of newly learned concepts.

It was considered important during the first session to scaffold the tasks in such a way that students had to complete simple tasks before moving onto more complex ones. This enabled them to build their knowledge directly on the concepts of previous tasks. Completion of the tasks was relatively unstructured.

Week 1. The students were asked to assemble a Magic Cube (see Figure 2A), followed by completing 8 discovery-based tasks (see Figure 2B), which were aimed at introducing the functionalities of the cubes' hardware components–sensors, actuators, Bluetooth connectivity, and how these components worked together. This number was chosen to allow the students many opportunities to succeed in order to foster a sense of accomplishment.
Week 2. The students created slide presentations about their first experience with the Magic Cubes. This was done as a way of encouraging the students to reflect on the concepts they had learned.
Week 3. The students learned to program the Magic Cubes using the ArduBlock programming environment [43] (see Figure 2C). This was designed to enable the students to move from understanding the *functionality* of the embedded hardware in the cube, and to being able to *control* the hardware components through programming. The task was segmented into a number of steps that were scaffolded in terms of conceptual complexity.
Week 4. The students were asked to design and create a paper prototype of their own "Internet of Things" device, by using their knowledge of sensors, actuators and wireless connectivity. This enabled the students to creatively apply their understanding of physical hardware functionality.
Week 5. The students were asked to program their own animations on the LED matrix of the cube using ArduBlock. This more open-ended activity was designed to enable the students to further their knowledge of writing algorithms and to additionally learn about writing *for loops* and creating *bitmaps*.
Week 6. The students were asked to conduct video interviews with their partners to ask each other about their overall experiences [24]. This assessment method was selected to enable the children to voice their perceptions about their experience during the 6 weeks, and discuss what was fun, interesting, difficult or boring for them

Table 2. Details of activities and rationale for the 6 sessions

Data Collection and Analysis

During each session, continuous audiovisual data was collected of the students' dialogue and interactions with each other and with the materials provided. Placement of multiple cameras throughout the room ensured that both the students' interactions in groups and the overarching classroom interactions (i.e., between groups, and between the students and instructors) were continuously visible. The researchers also wrote field notes. In the final session, the students conducted peer interviews with each other about their subjective experiences, and the researchers interviewed the teacher about the five prior sessions. This was done to provide multiple perspectives of the students' engagement, learning outcomes, and overall experiences.

The analysis of the audiovisual data was done using Interaction Analysis (IA), a qualitative analytic method that assumes interaction and knowledge are fundamentally situated in social and material ecologies [18]. It was chosen because of its suitability for the 'in the wild' approach [30] adopted in the research to study students' interactions with the Magic Cubes in the social and techno-material context of their classroom. Through a number of data sessions, two to three researchers, who had all been present in the sessions, first discussed the field observations and watched segments of video together. To aid the analysis, content logs were created based on the students' interactions and dialogue in the videos, and annotations added to index where the observed phenomena occurred in the social and temporal context of the tasks. Through collaborative discussion between the researchers, observed events were categorised into themes based on recurring instances. These were then refined with the students' and teachers' perceptions of their learning, as identified through the interviews.

Constructs of Analysis

The focus of analysis was to describe how the Magic Cubes and the associated task types could best support three key aspects of learning that SEN students are often said to need additional support in, and that tangible and physical interfaces have been suggested to support: *collaboration*, *comprehension* and *engagement.* In analyzing collaboration, we draw from Roschelle and Teasley's perspective that collaborative learning entails the 'continued attempt to construct and maintain a shared conception of a problem' [31]. Through this lens, we analyzed whether and how each student was able to support the learning of others, by physically sharing the technology, instructing their partner and reinforcing others' learning through dialogue. In analyzing comprehension, we chose to examine how engaging with the technology and learning tasks led to the students' reflection on the target learning concepts [1]. Hence, we analyzed comprehension more as a process, rather than an outcome. The analysis focused on dialogue between the students and instructors, indicating comprehension, or conversely, dialogue that indicated lack of understanding. When analyzing engagement, we were motivated by Price and Falcão's framework [25], which characterizes how different foci of attention all interplay during the learning process—for example, focus of attention on the technology, on tangential activities, and on the explicit learning outcomes. In our analysis, we examined the strategies the students used to regulate their attention to the technology and the learning tasks, and what aspects of the learning task made this easier or more difficult to do.

FINDINGS

Below, we describe the key findings from our analysis of the students' interactions—both with the interface and with each other—during the learning tasks, illustrating the IA themes through a selection of representative vignettes.

Collaboration

In analyzing collaboration, we define the term as working together to complete learning tasks. We characterize collaborative activity in terms of: individuals in pairs sharing control of task-related materials, visually attending to each other's actions, and verbally discussing the task. In our analysis, we examine a) collaborative trends for each task, across pairs, and b) pairs' collaboration patterns throughout the intervention.

Overall, the majority of students were seen to actively collaborate on all of the making, discovery-based and programming tasks. However, the nature of the collaboration that took place was qualitatively different between learning tasks. Next, we present in detail how patterns of collaboration evolved throughout the sessions.

Fluid collaboration in unstructured, exploratory tasks

The making and discovery-based tasks in week 1 were carried out using only the physical interface, without a desktop computer. In this session, the students were sitting around two circular tables, and engaging with tasks that called on unstructured, embodied exploration of the Magic Cubes. Collaboration within pairs appeared to be fluid, in the sense that the students in each group frequently watched and mimicked the others. The students were seen to take turns exploring the cubes' functionalities and discussing the hidden effects together. In particular, when new discoveries were made of the hidden sensor effects instantiated in the cubes, the students explicitly shared their cubes with their partners, by showing each other how the sensor effects worked, handing the cubes over, and instructing each other. This trend occurred across all pairs.

It was observed that in week one, collaboration also occurred frequently between pairs. For example, in the task in which the students first put together the Magic Cubes, the students were not told how the cubes would function once they were assembled. After two students, Teddy and Neil (G4), finished assembling the cube, Teddy was quietly told by one of the instructors to *"try shaking it"*. As he did this, the light inside the cube turned on for the first time. Two nearby students, who were looking at Teddy, exclaimed *"wow!"*, which in turn led to everyone at the table looking towards Teddy's cube. Instantly, all three pairs sitting at the table started shaking their cubes.

Because the eight discovery-based tasks were designed to be self-paced, the pairs around the table were often working on different tasks at any given moment in time. Nevertheless, between pairs, the students were seen to visually attend to each other's discoveries, in particular when someone in another group verbally called attention to their discovery. For example, Teddy, who was one task ahead of Lily and Gary (G5), discovered a sensor effect that entailed blowing hot air into the cube's temperature sensor in order to produce a growing fire animation on the LED matrix. When he successfully elicited the fire animation, he exclaimed *"hey look, I made fire!"*, pointing the LED matrix toward Lily, who responded *"oh, cool!"*. It was observed that once Lily and Gary moved to this discovery task, they immediately copied the action they had previously observed Teddy doing, without testing any other actions on the cube, suggesting that they had implicitly learned the sensor effect by observing Teddy's actions.

Static collaboration and division of labor in programming tasks

Collaboration patterns both within and between groups were qualitatively different during the programming tasks, in which the students were sitting in rows and facing computer screens, rather than at circular tables without computers. Within groups, the students implicitly divided their roles when collaborating. Specifically, in most pairs, one student held the instruction sheet and read aloud the step-by-step instructions, while the other controlled the programming software. This may have been because it was more convenient for one student to consistently access the keyboard and mouse than to share control. In all except one group (G4), the students were seen to point to the screen throughout the learning task, and to discuss where to place the programming blocks in the programming environment.

During the programming tasks, the collaboration between groups was less frequent. The students periodically observed the actions of those around them, but their visual attention was predominantly on the computer screen used within their group. When observation of other groups occurred, this was most often tied to "loud" events in which the other group verbally expressed excitement after they had uploaded their code to the Magic Cube, or physical events in which the other group was shaking, or otherwise manipulating their Magic Cube in space. An example of this was a pair successfully uploading their "night light" code in week 3, and calling the teacher over to show off what they had achieved, then subsequently reaching the cube toward the ceiling light. At these points in time, the students in their proximity looked over toward their peers, and provided them with positive reinforcement (e.g., *"oh, wow!"*). However, in the programming activities, observing the end result (i.e., the program uploaded to the cube) did not help with the process of programming per se, so it is unlikely that this helped the onlookers with the programming process, other than possibly helping them understand the intended result of the program. This contrasts with the purely physical, discovery-based tasks in week 1, where observation of other, successful groups helped the students to complete the tasks.

During observed instances of talk between groups while working on the programming tasks, it was found that there were no cases of spontaneous sharing of code, or of discussing the programming concepts. Instead, the students mainly relied on the instructors, rather than their peers, for support with the programming. However, when prompted by the instructors, the students readily helped other groups. For example, in one programming session, Curtis and Fabian (G3), who were ahead of the others, were encouraged by one of the instructors to walk over to Lily and Gary (G5) and explain to them how to make two images display on the LED matrix in sequence, in order to create an animation. Curtis verbally instructed Lily and Gary on how to put blocks together in the programming environment in order to create an animation. In doing so, he led them through the trial and error process that he and Fabian had previously followed when trying to understand the concepts of sequences and delays. Specifically, he told Lily and Gary to program two images in sequence and upload the code. When they did so, the LED matrix on the cube began to flash rapidly. Curtis then explained why this was happening saying, *"that's why it looks so red… cause it's going so fast"*. He explained that they needed to add "delay statements" after each image in order to instruct the cube for how long to display each image. Lily asked him to clarify where the delay statements should go. Once Curtis confirmed that they had formatted the code correctly, Lily and Gary then started to independently experiment with the delay variable values, while Curtis and Fabian watched.

Unprompted support of the other within groups

Throughout the intervention, the students were often seen to actively help each other out within groups by capitalizing on each other's strengths. For example, in the programming tasks, David (ASD, G2) took the role of reading out instructions to his partner, Eric (Specific Speech and Language Difficulties), who had substantial challenges with reading. Similarly, Curtis (ASD, G5) read out the instructions to his partner Fabian, who was not fluent in English, while Fabian controlled the mouse and keyboard. In the discovery-based tasks, the students worked to come to the same level of understanding, when collaborating. For example, Jason (G1) was more active in exploring the cube, and often faster than his partner, Keith. However, Jason actively helped Keith to understand the concepts before the pair moved onto the next task. In one instance, when Jason had discovered an effect related to Bluetooth connectivity, which Keith had not, he showed Keith how to elicit the effect, while verbally explaining how it worked. The two then elicited the effect together by sharing control of the cube, resulting in them sharing a high five.

Sharing successes

A finding throughout the sessions was that the students consistently shared their successes with others after completing the tasks. Students who had successfully completed discovery-based tasks, or uploaded a new program to the cube, often drew attention from nearby peers, especially through verbal exclamations (e.g., *"I got it!"*). Moreover, they often stopped instructors who were walking past, in order to show off their discoveries, for example, by waving a cube in the air. Such instances were often met with positive feedback from their peers (e.g., *"cool!"*), and praise from the instructors (e.g., "*well done!*"). These moments were facilitated by the form factor of the cube making it easy to show off to others, for example, by waving the cube in the air and by tilting it toward someone on the other side of the table.

Breakdowns in collaboration

The exception to the collaboration patterns observed within groups was Neil (ASD, G4) and Teddy (ASD, G4). According to the class teacher, Teddy is normally able to grasp concepts quickly, but struggles with maintaining joint attention and often *"does his own thing"* during class lessons. Neil does not often verbally communicate, and it is often unclear whether or not he is actively attending to the class activities. In week 1, for the first thirty minutes of the exploratory making- and discovery-based tasks, Neil and Teddy were seen to both collaboratively engage with the learning tasks while sharing a cube. In particular, the pair was observed to mimic each other's actions when trying to elicit colors and animations on the cubes. However, midway through the session, Neil became disengaged and withdrew from actively taking part in further tasks. It was observed, however, that he was still visually focused on what others were doing, and filled out the worksheet appropriately when Teddy discovered the sensor effects. However, he did not pick up the cube himself, or test out the effects that Teddy had discovered, unless prompted by one of the instructors. Teddy continued to engage in collaborative activity with others nearby, even while his partner was disengaged from the tasks, such as discussing and sharing his insights on the sensor effects. The class teacher noted that this behavior surprised him, in a positive way, given his previous experience with Teddy.

In the programming-based tasks, Teddy was seen to take control of both the instructions and the computer keyboard and mouse, while Neil was disengaged from the programming tasks. No discussion took place within the pair, although Teddy often called over one of the instructors to ask questions. When asked by an instructor if he wanted to have a go at helping with the programming, Neil replied that he did not. It could be that the physical nature of the tasks in week 1, where no desktop computers were used and no static division of labor with a partner was required, made it easier for Neil to participate in collaborative activity. However, because he gave very short and off-topic responses in the peer interview when asked about his experiences with programming, it is unclear why he was disengaged in the latter sessions.

Comprehension

In this section, we report on how comprehension of the computing concepts unfolded as the students interacted with the interface, instructions, and programming environment during the learning tasks.

Instructions and instructors in open-ended exploration

In week 1 of using the toolkit, the instructions were given only verbally. Visual task sheets were provided for the students as a supplement to the verbal instructions, and to enable the students to easily write down their discoveries. The lack of explicit, written instructions was seen to be effective for encouraging open-ended exploration, as supported by evidence of all the students trying out a variety of physical actions (e.g., tilting, shaking, covering, blowing) on the sensors. However, simultaneously, because of the lack of step-by-step instructions, when the students failed to discover a particular effect and got "stuck", the role of the instructor became crucial in enabling them to move forward in the task. Specifically, in these cases, when the instructors noticed that a pair was struggling, they would approach the students, and give them hints about how to proceed with the task, without giving away the answers. Because of the small class size, the students were able to receive help, and quickly continue with the tasks.

Instructions and instructors in programming tasks

In the programming tasks in weeks 3 and 5, written step-by-step programming instructions were provided. These were also supplemented with images showing how the block-based code should look at each step in the ArduBlock programming environment. This was done in order to support the students who had difficulties in reading, and make it easier for the students to self-monitor their progress. It was observed that the majority of groups engaged with the written instructions; these groups read the instructions aloud, and verbally discussed and pointed to where the code should go in the programming environment. This was seen to have helped the pairs to form expectations of what the intended result of the code should be. For example, David and Eric (G2), who discussed the instructions during the "night light" task at length, had an expectation of how their program should function before uploading it to their cube. When asked by an instructor what they thought it should do, before testing it, Eric stated: "*the light will turn on and off, with the light level"*. Immediately after uploading the code, he proceeded to demonstrate this by covering the light sensor on the cube, without expressing surprise.

Two groups, however, relied predominantly on the visual images in the instruction sheets (G4 - Teddy and Neil, and G5 – Lily and Gary). Here, instructors played a key role in helping the students to move past blocks in their understanding. For example, when he noticed that they were struggling with the written instructions, one of the key workers helped Lily and Gary by reading the instructions to them out loud. Additionally, it was observed that Teddy did not focus his attention on the written instructions during the "night light" task, and neither read them aloud, nor heard them being read by others. Because of this, it is likely that he completed the task without reflecting on the concepts. Once he uploaded the program, he did not understand what the intended effect on the cube should be. At this stage, he required support from an instructor to explain both the program and how it manifested on the cube.

Verbally reflecting

The process of sharing successes and showing off what was accomplished engendered an evident sense of achievement and pride in the students. In addition, it was seen to serve a functional role in probing active reflection. Specifically, when the students shared their successes with the instructors, this enabled the instructors to ask them to explain what they had discovered or programmed. In many instances, this elicited verbal reflection, and enabled them to clarify their understanding. For example, one instructor approached Jason and Keith (G1) during a discovery-based task. Jason quickly said, "*I figured it out. It is movement",* referring to the sensor that caused the light inside the Magic Cube to turn on. He and Keith demonstrated this, by shaking the two cubes at the same time. However, the instructor saw that they were missing a key aspect of the task—that the two cubes were interconnected through Bluetooth, and when both were being shaken simultaneously, the color of the neopixel light was different than when only one was being shaken. The instructor asked them to try shaking only one cube at a time, and then both cubes simultaneously. They then quickly understood the effect, and Jason exclaimed, *"It's going purple! So, the two colors together – they make purple".*

Understanding computational concepts through embodied interaction

The making and discovery-based tasks in week 1 were designed to capitalize on embodied interaction, where the tasks could only be successfully completed by shaking, tilting, and blowing into the Magic Cubes. The first week's session, therefore, enabled the students to build their knowledge by using their bodies to explore concrete examples related to abstract computing topics (i.e., the functionality of sensors and actuators, and connectivity between cubes). The students were seen to also use the physical properties of the cubes to clarify their understanding during the programming tasks. Most groups used the cubes, alongside their code, to iteratively refine their understanding of the programming concepts through embodied interaction. For example, in the "night light" programming task, Curtis (G3) was unsure if the code he had uploaded was behaving as it was supposed to. The instructor asked him to verbally walk through what his expectations were, based on the instructions he had read. As he did so, he used the cube to physically trace whether the program statement was working as expected. As he turned the light sensor side of the cube toward the light, he said *"it turns off"*. He then proceeded to turn it toward the floor, tilting his body toward his partner and saying, "*and now if you point it towards there, it's lighting up… so it makes sense*". Hence, the cubes enabled a concrete, physical representation of the program through which the students were able to use their existing knowledge of the physical world to test hypotheses and refine their understanding.

Engagement

Our lens of analysis when describing the students' engagement during the sessions focused on how the learning tasks mediated sustaining and switching of attention and focus during the learning process.

Self-paced session structure

The self-paced structure of the sessions enabled the students to proceed with the tasks at their own speed, without rushing to catch up with the rest of the classroom. In addition, in this set-up, the students were able to ask for individual help and clarifications from the instructors, which enabled the wide variety of abilities in the classroom to be supported. The tasks were conceptually scaffolded in a way so that if they did not complete them all in one session, it did not affect the ability to proceed with new tasks in the following week. This proved to be an effective strategy, as there were no observed instances of students rushing to finish a specific task – rather, they were seen to take their time in exploring the interface, and experimenting with their code.

In addition, the self-paced structure allowed the students to self-regulate their focus on the task. For example, at one point, Teddy (ASD) completed a programming task. Near the end of it, he seemed uneasy, as indicated by him moving in his seat more than usual, and looking around the room for an extended period of time, without focusing on the task-related materials. When an instructor noticed this, she asked Teddy if he would like to start on the next task. He replied that he would not, and decided to take a break from the activity. He chatted with his peers nearby, and went online to look up some tunes. Five minutes later, when a pair sitting next to him started the next task, he decided to also join in and start again. He was once again highly focused on the programming. The self-paced task, therefore, enabled the students to decide when they needed a break from the activities. In Teddy's case, it allowed him to regulate his focus himself, rather than be forced to stay engaged for a consecutive hour and a half.

The relationships between difficulty, enjoyment and engagement

It was observed that there was a correlation between the difficulty of the task, and how focused and consistently engaged the students were seen to be when completing it. For example, in week 1, after the students had assembled the Magic Cube, they were asked to carry out a particularly difficult task which entailed drawing three-dimensional shapes in the air with the cubes in order to produce various colors of light. It was difficult for most of the students to get the colors to work. However, they persisted in trying for a long time. They took turns trying to draw the shapes with their partners, and clapped when others around them managed to get the colors to work. The challenging element of the activity seemed to add to the anticipation and suspense of eliciting the intended effects, in turn sustaining their focus. In the peer interviews, several of the students said that this was one of their favorite tasks. Conversely, when tasks became too difficult or ambiguous, where it was unclear how to proceed, the students often became stuck and disengaged. Here, the role of the instructors was integral to getting them back on track, by providing individualized support.

DISCUSSION

Our findings have shown, at a fine level of granularity, how a physical toolkit for learning computing can be effectively introduced to a mixed SEN classroom. Moreover, we found that a physical toolkit could be used not only to support comprehension of computational concepts, but also to enable students to get excited about learning, and to inclusively engender collaborative and engaging experiences. The waves of collaborative interaction throughout the classroom, enabled by the physicality of the toolkit, helped the students to support each other's strengths when learning together. Furthermore, the design of the sessions enabled most of the students to sustain focus on the tasks for extended periods of time, which was surprising even to the class teacher.

Our findings suggest that physical interfaces can be introduced to good effect in mixed SEN classrooms when teaching computing. However, it is not enough to assume they can be designed as off-the-shelf toolkits for SENs. A number of other factors need to be considered to ensure that the learning tasks they are used with are designed to be inclusive to all students. These are: (i) enabling collaborative learning, (ii) enabling embodied debugging, (iii) supporting self-regulated and scaffolded learning, (iv) providing instructions in multiple representations, and (v) taking advantage of informal assessments.

(i) Enabling collaborative learning

Our findings showed that the hand-sized shape of the Magic Cubes interface, paired with their highly visible effects, allowed for a diversity of shared interactions to occur. The students used the physical affordances of the cubes to observe the activities of others, and to readily show off their successes to their peers and the instructors. In particular, when completing purely physical tasks where no computer screen was involved, this led to collaboration both *within* and *between* pairs, where the students were seen to learn together by watching and mimicking others.

Based on these findings, when collaboration between peers is the goal, there appears to be much potential for using physical interfaces to teach abstract concepts. To date, many studies have also pointed to the ability for physical and tangible interfaces to foster higher levels of collaboration in general (e.g., [14,17]). However, these effects are often attributed to the form factor of the interface itself, without detailed explanation of the effect of the task type and materials on interaction. Our study suggests that the classroom set-up and use of other materials, such as desktop computers, are also important factors at ensuring and promoting collaborative learning.

(ii) Enabling embodied debugging

The students were seen to use their bodies to understand the concepts instantiated in the discovery-based and programming tasks. This suggests that having physical tasks can facilitate learning about abstract functionalities of sensors, actuators, and about programming constructs through enabling students to enact them out. Supporting mental "debugging" through embodied actions has long been suggested to assist learning, stemming back to Papert's seminal turtle LOGO, in which children programmed a physical turtle to learn geometry concepts [21]. Where it has been suggested that kinesthetic and embodied learning is particularly important for students with intellectual disabilities [11], we suggest that designing tangible and physical interfaces that support embodied "debugging" can also be a compelling way to inclusively teach abstract concepts related to computing in SEN.

(iii) Supporting self-regulated and scaffolded learning

Although previous research has suggested that structured learning activities may be more appropriate for students with learning difficulties [11], we found that the open-ended and self-paced design of learning tasks in our intervention was effective in promoting inclusive learning for a variety of abilities and needs. The self-paced structure of the tasks enabled students with a wide spread of abilities to stay engaged in the learning activities. They were able to decide at which speed to complete the tasks, and to decide when to take breaks when they had had enough. Moreover, using shorter tasks in the sessions, together with conceptually scaffolded levels of complexity, meant that there was no time pressure on the students to finish at the same time as others. In addition to reducing stress on the students, this enabled the instructors to provide targeted and individualized support to small groups of students when needed, rather than constantly addressing the class as a whole and working to ensure all students are simultaneously at the same point in the tasks. Where typical SEN classrooms usually have small class sizes and more than one instructor, this strategy of using short, self-regulated tasks in interventions can carry over to other interventions in SEN.

(iv) Providing instructions in multiple representations

An important finding was that engagement and comprehension were contingent on the students receiving instructions appropriate to their needs. To this end, we provided the students with a mix of verbal, visual and written instructions. Depending on their abilities and strengths, the students were seen to use one or a mix of these when completing the tasks. This enabled them all to complete the tasks. However, students who relied on the purely visual, step-by-step instructions—such as photos representing the intended code structure in the programming tasks—were seen to reflect less on what was *being done* than those who relied on the written and verbal instructions. When the students relied purely on the visual instructions, they were able to complete the tasks but then often not able to understand the effects embedded in the cubes. In these instances, the instructor had to step in and provide individualized support. Given the importance of providing appropriate instructions in SEN [11], we suggest future work investigate the interplay between learning interfaces and instruction representation in more detail.

(v) Taking advantage of informal assessments

Using a variety of types of informal assessments was found to help the students consolidate their understanding of computing concepts, in particular through using their creativity. These included, specifically, slide presentations in week 2, a design and paper prototyping challenge in week 4 and peer interviews [24] in week 6. The students all stated that they enjoyed creating artifacts as methods of assessment. These methods also helped us get a sense of the students' comprehension and from this to adapt the learning tasks planned for following weeks. Our findings suggest that they can help provide estimates of students' comprehension while contributing positively to the overall experience of the intervention.

CONCLUSION

There can be many challenges for supporting learning in SEN classrooms, especially for abstract topics like computing. Students often have a wider mix of abilities than their peers in mainstream school settings, and it can be difficult to structure learning tasks that simultaneously provide engaging and effective learning experiences for all. However, as our study has shown, the affordances of physical interfaces have much promise in SEN classrooms, especially when the design of the task type and supporting materials enable self-regulated, embodied learning with appropriate support from the instructors. If tasks are designed in this way, physical interfaces can enable students with a range of difficulties to leverage their abilities to collaborate and engage with curricular content, while fostering comprehension, enjoyment and a sense of self-accomplishment. There is much scope for designing new technologies to support more inclusive computing.

ACKNOWLEDGMENTS

This research has been made possible by the EPSRC and the BBC through an iCASE scholarship (Award Number 1623937). We would like to give a special thanks to Rory Matthews, the key workers, and, most of all, the students who participated for their time, support, and enthusiasm.

REFERENCES

1. Edith K. Ackermann. 1996. Perspective-Taking and Object Construction: Two Keys to Learning. In

Constuctionism in Practice: Designing, Thinking, and Learning in a Digital World (Kafai, Y., and Resnick, M., Eds.). Mahwah, New Jersey: Lawrence Erlbaum Associates. Part 1, Chapter 2., 25–37.

2. Alissa N. Antle. 2009. LIFELONG INTERACTIONS Embodied child computer interaction: why embodiment matters. *interactions* 16, 2: 27–30.
3. Arduino. 2016. Open-source electronic prototyping platform. Retrieved from https://www.arduino.cc
4. BBC micro:bit. 2016. Micro:bit: get creative, get connected, get coding. Retrieved from https://www.microbit.co.uk
5. Ayah Bdeir. 2009. Electronics as material: littleBits. In *Proceedings of the 3rd International Conference on Tangible and Embedded Interaction*, 397–400.
6. Peter Börjesson, Wolmet Barendregt, Eva Eriksson, and Olof Torgersson. 2015. Designing Technology for and with Developmentally Diverse Children: A Systematic Literature Review. In *Proceedings of the 14th International Conference on Interaction Design and Children* (IDC '15), 79–88. https://doi.org/10.1145/2771839.2771848
7. Karen Brennan and Mitchel Resnick. New frameworks for studying and assessing the development of computational thinking. Retrieved September 6, 2017 from http://scratched.gse.harvard.edu/ct/files/AERA2012.pdf
8. CodeMe. 2016. The CodeMe toolkit. Retrieved from http://www.codeme.io
9. Department for Education. 2017. Special educational needs in England: January 2017. *Department for Education*. Retrieved from https://www.gov.uk/government/uploads/system/uploads/attachment_data/file/633031/SFR37_2017_Main_Text.pdf
10. Paul Dourish. 2004. *Where the Action is: The Foundations of Embodied Interaction*. MIT Press.
11. Taciana Pontual Falcão and Sara Price. 2010. Informing design for tangible interaction: a case for children with learning difficulties. In *Proceedings of the 9th International Conference on Interaction Design and Children*, 190–193.
12. William Farr, Nicola Yuill, Eric Harris, and Steve Hinske. 2010. In my own words: configuration of tangibles, object interaction and children with autism. In *Proceedings of the 9th International Conference on Interaction Design and Children*, 30–38.
13. Samantha Holt and Nicola Yuill. 2014. Facilitating other-awareness in low-functioning children with autism and typically-developing preschoolers using dual-control technology. *Journal of autism and developmental disorders* 44, 1: 236–248.
14. Michael S. Horn, R. Jordan Crouser, and Marina U. Bers. 2012. Tangible Interaction and Learning: The Case for a Hybrid Approach. *Personal Ubiquitous Comput.* 16, 4: 379–389. https://doi.org/10.1007/s00779-011-0404-2
15. Michael S. Horn, Erin Treacy Solovey, R. Jordan Crouser, and Robert JK Jacob. 2009. Comparing the use of tangible and graphical programming languages for informal science education. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, 975–984.
16. Maya Israel, Quentin M. Wherfel, Jamie Pearson, Saadeddine Shehab, and Tanya Tapia. 2015. Empowering K–12 students with disabilities to learn computational thinking and computer programming. *TEACHING Exceptional Children* 48, 1: 45–53.
17. Rose Johnson, Venus Shum, Yvonne Rogers, and Nicolai Marquardt. 2016. Make or Shake: An Empirical Study of the Value of Making in Learning About Computing Technology. In *Proceedings of the The 15th International Conference on Interaction Design and Children* (IDC '16), 440–451. https://doi.org/10.1145/2930674.2930691
18. Brigitte Jordan and Austin Henderson. 1995. Interaction analysis: Foundations and practice. *The journal of the learning sciences* 4, 1: 39–103.
19. Yasmin B. Kafai, Quinn Burke, and Mitchel Resnick. 2014. *Connected code: Why children need to learn programming*. Mit Press.
20. Paul Marshall. 2007. Do Tangible Interfaces Enhance Learning? In *Proceedings of the 1st International Conference on Tangible and Embedded Interaction* (TEI '07), 163–170. https://doi.org/10.1145/1226969.1227004
21. Seymour Papert. 1980. *Mindstorms: Children, computers, and powerful ideas*. Basic Books, Inc.
22. Jean Piaget. 2013. *The construction of reality in the child*. Routledge.
23. Taciana Pontual da Rocha Falcão. 2014. Discovery learning with tangible technologies: the case of children with intellectual disabilities. UCL Institute of Education. PhD Thesis.
24. Dylan J. Portelance and Marina Umaschi Bers. 2015. Code and Tell: Assessing young children's learning of computational thinking using peer video interviews with ScratchJr. In *Proceedings of the 14th International Conference on Interaction Design and Children*, 271–274.
25. Sara Price and Taciana Pontual Falcão. 2011. Where the attention is: Discovery learning in novel tangible environments. *Interacting with Computers* 23, 5: 499–512.
26. Mitchel Resnick. 2006. Computer as paint brush: Technology, play, and the creative society. *Play=*

learning: How play motivates and enhances children's cognitive and social-emotional growth: 192–208.

27. Mitchel Resnick. 2013. Learn to code, code to learn. *EdSurge, May*. Retrieved September 6, 2017 from https://www.edsurge.com/news/2013-05-08-learn-to-code-code-to-learn
28. Mitchel Resnick, John Maloney, Andrés Monroy-Hernández, Natalie Rusk, Evelyn Eastmond, Karen Brennan, Amon Millner, Eric Rosenbaum, Jay Silver, Brian Silverman, and others. 2009. Scratch: programming for all. *Communications of the ACM* 52, 11: 60–67.
29. Mitchel Resnick, Fred Martin, Robert Berg, Rick Borovoy, Vanessa Colella, Kwin Kramer, and Brian Silverman. 1998. Digital manipulatives: new toys to think with. In *Proceedings of the SIGCHI conference on Human factors in computing systems*, 281–287.
30. Yvonne Rogers and Paul Marshall. 2017. Research in the Wild. *Synthesis Lectures on Human-Centered Informatics* 10, 3: i-97. https://doi.org/10.2200/S00764ED1V01Y201703HCI037
31. Jeremy Roschelle and Stephanie D. Teasley. 1995. The Construction of Shared Knowledge in Collaborative Problem Solving. In *Computer Supported Collaborative Learning*, Claire O'Malley (ed.). Springer Berlin Heidelberg, 69–97. https://doi.org/10.1007/978-3-642-85098-1_5
32. Sue Sentance, Jane Waite, Steve Hodges, Emily MacLeod, and Lucy Yeomans. 2017. "Creating Cool Stuff": Pupils' Experience of the BBC Micro:Bit. In *Proceedings of the 2017 ACM SIGCSE Technical Symposium on Computer Science Education* (SIGCSE '17), 531–536. https://doi.org/10.1145/3017680.3017749
33. Sowmya Somanath, Lora Oehlberg, Janette Hughes, Ehud Sharlin, and Mario Costa Sousa. 2017. "Maker" Within Constraints: Exploratory Study of Young Learners Using Arduino at a High School in India. In *Proceedings of the 2017 CHI Conference on Human Factors in Computing Systems* (CHI '17), 96–108. https://doi.org/10.1145/3025453.3025849
34. Hideyuki Suzuki and Hiroshi Kato. 1995. Interaction-level Support for Collaborative Learning: AlgoBlock—an Open Programming Language. In *The First International Conference on Computer Support for Collaborative Learning* (CSCL '95), 349–355. https://doi.org/10.3115/222020.222828
35. Anja Thieme, Cecily Morrison, Nicolas Villar, Martin Grayson, and Siân Lindley. 2017. Enabling Collaboration in Learning Computer Programing Inclusive of Children with Vision Impairments. In *Proceedings of the 2017 Conference on Designing Interactive Systems* (DIS '17), 739–752. https://doi.org/10.1145/3064663.3064689
36. UK Department of Education. 2016. National curriculum in England: design and technology programmes of study - GOV.UK. Retrieved from https://www.gov.uk/government/publications/national-curriculum-in-england-design-and-technology-programmes-of-study/national-curriculum-in-england-design-and-technology-programmes-of-study
37. Marjo Virnes, Erkki Sutinen, and Eija Kärnä-Lin. 2008. How Children's Individual Needs Challenge the Design of Educational Robotics. In *Proceedings of the 7th International Conference on Interaction Design and Children* (IDC '08), 274–281. https://doi.org/10.1145/1463689.1463766
38. Lev Semenovich Vygotsky. 1978. *Mind in society: the development of higher psychological processes*. Harvard University Press.
39. Jeannette M. Wing. 2006. Computational thinking. *Communications of the ACM* 49, 3: 33–35.
40. Oren Zuckerman, Saeed Arida, and Mitchel Resnick. 2005. Extending tangible interfaces for education: digital montessori-inspired manipulatives. In *Proceedings of the SIGCHI conference on Human factors in computing systems*, 859–868.
41. Oren Zuckerman and Ayelet Gal-Oz. 2013. To TUI or not to TUI: Evaluating performance and preference in tangible vs. graphical user interfaces. *International Journal of Human-Computer Studies* 71, 7: 803–820.
42. Raspberry Pi - Teach, Learn, and Make with Raspberry Pi. *Raspberry Pi*. Retrieved from https://www.raspberrypi.org/
43. Ardublock | A Graphical Programming Language for Arduino. Retrieved from http://blog.ardublock.com/